

6. Fabiana

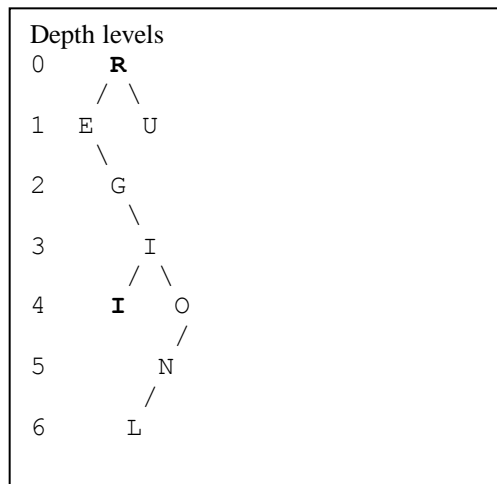
Program Name: Fabiana.java

Test Input File: fabiana.dat

Fabiana has just learned about the depth concept as it relates to binary search trees and decides to do some research. She takes a single word, picks a letter from somewhere in the word, and finds the depth of where that letter settles into a binary search tree. Doing it by hand is not difficult, but is time consuming, and she needs your help in writing a program to do it for her to save her some time doing the research.

Given a single word with no spaces, followed by an integer P after a blank space, output the depth at which the letter in position P is inserted into a binary search tree.

Below is an example of a binary search tree formed by using the letters in the word REGIONUIL. The first letter is always the root of the tree, and then each subsequent letter "finds" its place relative to the root, and any other nodes in the tree, going to the left if its value is less than or equal to the current node, and going right if it is greater in value.



The root node that contains the letter R has a depth of zero, with the letters E and U at level 1, G at level 2, and so on.

The data set consisting of the word REGIONUIL and the value 0 means to find where the letter at position 0, the R, settles in the tree, which would be the root, or level 0.

The data set REGIONUIL 7 indicates the second occurrence of the letter I, which is inserted into the tree at level 4 as shown.

Input: A series of data sets, each on a separate line, consisting of a word (all uppercase) and an integer.

Note: The judges test data input will have different words than the one given here.

Output: The depth (or level) at which the indicated letter in the word settles in the binary search tree.

Sample Input:

```

REGIONUIL 0
REGIONUIL 1
REGIONUIL 3
REGIONUIL 5
REGIONUIL 7
    
```

Sample Output:

```

0
1
3
5
4
    
```